

Amendments to the Drawings

Please replace the drawing sheets currently on file, with the replacement drawing sheets containing an amended set of Figures 1-12 submitted herewith.

REMARKS

Applicant wishes to thank the Examiner for reviewing the present application.

Amendments to the Specification

The specification is amended on page 10 to correct typographical errors. No new matter is added by way of these amendments.

Amendments to the Claims

Claim 4 is amended to add further technical language regarding the hardware used by the framework. Claim 4 is also amended to include an interface, process selector, data selector and storage device. The interface, data selector and process selector were formerly the subject matter of claims 5, 16 and 17 respectively. Accordingly, claims 5, 16 and 17 are canceled.

Claims 6 and 8 are amended to correct their dependencies, consistent with the amendments to claim 4.

Claim 20 is amended in accordance with the cancellation of claim 5, and the amendments to claim 4. Claim 24 is amended to correct a typographical error.

The method of claim 33 is amended for consistency with the framework of claim 4, which included adding the subject matter of claim 34. Accordingly, claim 34 is cancelled.

Claims 35 and 48 are amended to correct their dependencies, consistent with the amendments to claim 33, and claim 46 is amended similar to claim 20.

No new subject matter is believed to have been added by way of these amendments.

Amendments to the Drawings

Replacement drawings are submitted in accordance with 37 CFR 1.121(d). The new drawings are clear, of the proper format, and are believed to be acceptable. Applicant advises that Figures 11 and 12 are now shown in a single drawing sheet.

Claim Rejections – 35 U.S.C. §101

Claims 4-32 were rejected under 35 U.S.C. §101 for being directed to non-statutory subject matter. Claim 4 is amended to indicate that the framework recited therein is “computer implemented”, and that the framework monitors workflow within a “computer application”.

Moreover, claim 4 is amended to include a user interface, data storage device, a process selector, and a data selector. Accordingly, Applicant submits that the framework of claim 4, as amended, is executable in the context of “functional descriptive material”, and as such complies with 35 U.S.C. §101. Applicant advises that similar amendments have been made to claim 33 for consistency.

Claim Rejections – 35 U.S.C. §112

Claims 4-32 were rejected under 35 U.S.C. §112, second paragraph for being indefinite due to the expression “the applied processing” in claim 4. Claim 4 is amended to include a process definition, and the expression “the applied processing” is replaced by “the execution of said process definition”. Such amendment is believed to overcome the Examiner’s rejection, and claims 4-32 are believed to comply with 35 U.S.C. §112, second paragraph.

Claim Objections

Claims 4-32 were objected to for the uppercase “A” included in elements (a), (b), (c) of former claim 4. In amended claim 4, such uppercase letters have been replaced by their lowercase counterparts to overcome the Examiner’s objection.

Claim Rejections – 35 U.S.C. §102(e)

Claims 4-51 have been rejected under 35 U.S.C. §102(e) as being anticipated by U.S. Patent No. 6,606,740 to Lynn.

Claim 4 has been amended to include the features of former claims 5, 16 and 17, now cancelled. Claim 4 is also amended to further define the features of the framework, which includes the addition of a process selector, data selector, data storage device, user interface, and process definitions. Claim 4 is also amended to clarify the nature of the environment in which the framework is used, namely the preamble is amended to define the framework as being a computer implemented framework that is used for monitoring workflow within a computer application. These amendments are believed to clarify the distinctions in the claims with respect to the Lynn reference, as supported by the following discussion.

The present application relates to a framework for a computer implemented application, which seeks to build the application around multiple levels of functionality, and may or may not

be visible to the user of the framework. Three levels of functionality are provided as follows.

The first level is a process level, which may also be referred to as a “protocol”, as it encapsulates a set of data plus a definition of the processing steps to be applied. An example of a process instance is a study, which is being viewed in a 2D protocol. This level is managed by the framework and there can be multiple “processes” active at the same time, e.g., the user can be reviewing one study with one process, and at some point, switch to reviewing a study from a different patient (i.e. the framework manages the different process instances that may be active). In a simple case, a system may hide selection at this level, and present a single process to a user. A process includes a set of “activities”, with the possibility of automated control flow from one activity to the next, based on properties of the activities, and external rules.

The second level, is a sub-process level. Although this level is listed as the next level in the hierarchy, it is really just a grouping of activities. This grouping of activities is useful for reuse, convenience of navigation, and grouping activities with shared tools. Generally, a process will make activities visible to the end user, which may even need to have flexibility to switch the order of activities, based on something that is not pre-programmed logic. Thus, the sub-process level may present a user interface for navigating between activities, and for indicating the current position in a process.

The third level, is an activity level. This is the lowest level of the hierarchy that is explicitly supported by the framework (an activity can internally present a deeper model, if required). An activity will have external properties, which are altered as the result of processing, and can then be used to achieve sequencing of activities within a process. It is possible to dynamically (at run time) add or remove activities to a process, so that, for example, a process can provide different functionality based on results of earlier activities. Each activity can be implemented as a separate component.

The framework outlined above provides a means to monitor, author, interact, etc., a workflow within a computer application. Such workflow may be referred to as “desktop workflow”.

Claim 4 provides a framework within a computer application that can combine components from different sources to use in the application, and has a user interface, and three levels of functionality similar to what is described above. At the process level, a process definition can be selected, that defines a set of process steps to be applied to a data set. A

process selector is provided for selecting such a process definition, and a data selector is provided for selecting a data set from a group thereof. The groups of data sets and process definitions are stored in a data storage device that is accessible by the application.

Claim 4 also provides a sub-process level that is an aggregation of selected activities that are associated with a particular selected process definition. The sub-process level enables the navigation between the activities within the application during execution of the process definition. The activity level includes at least one activity that is part of a set of activities. The activities are selected from the activity level during the process definition and the selection thereof is performed at the process level. The activities have a property that is modified as a result of execution of the process definition that changes the data set to produce an output data set for the user interface provided by the framework.

The Lynn reference describes a workflow processing framework that provides common objects and business processes for the creation of an enterprise-wide workflow processing system. Conventional workflow, database, and other platforms are accessed by standard protocols. The framework described by Lynn uses a set of software objects, each uniquely performing a corresponding function in such an enterprise-wide workflow, and a set of process definitions, accessed by a subset of the software objects, to provide a flexible processing system.

Lynn's system is directed to coordinating workflow across an entire enterprise, and is particularly concerned with specific software objects that can coordinate and/or facilitate different software formats in different areas of the enterprise. In particular, Lynn teaches a level of business processes that are environment specific modules for implementing foundation objects through a mediating common objects level. This hierarchy is used to enable different formats from the foundation objects to be used by the business processes at the interactive level. Such workflow does not provide an application for interacting with particular process definitions, and is thus inherently different from what is recited in claim 4.

Applicant respectfully submits that Lynn does not teach several elements recited in amended claim 4, and is not even directed to a similar type of framework.

At the process level of claim 4, a data set can be selected using a data selector, and a process definition can be selected using a process selector. These selected items are chosen from particular groups that are stored by a data storage device. Therefore, the process level enables the user to interact with the other levels therethrough, enabling the selection of particular data

sets and process definitions as desired. As noted above, Lynn is concerned with enterprise-wide workflow, and thus is not concerned with a single application as recited in claim 4, but aims to provide seamless operation over several applications, without the user defining any interaction therebetween. Lynn aims to have the user unaware of the process, whereas in claim 4, the process is to be chosen as a process definition as it pertains to a data set. Such distinctions emphasize the inherent difference between desktop workflow (e.g. claim 4) and enterprise workflow (e.g. Lynn).

The sub-process level includes an aggregation of activities that are associated with a process definition. This enables the user to navigate between selected activities during execution of the process definition. Lynn's system is entirely unsuitable for such sub-processing. Lynn aims to inhibit transparent navigation between activities according to user selection (i.e. according to a process definition) by pulling different formats across an enterprise and providing them to the business process level without the user aware of any navigation across the enterprise. The Examiner equates Lynn's common functions (24) with claim 4's sub-process level. Applicant believes that these features are fundamentally different. Although each of these levels essentially serve to coordinate functionality between other levels in the respective frameworks, they operate, and ultimately achieve a different result.

The activities recited in claim 4 have properties that are modified as a result of the execution of the process definition. The properties are applied to the data set to provide an output data set. In claim 4, the user selects the data set, and the process definition for modifying the data set, i.e. the workflow. In Lynn's system, the user operates at the business level to perform individual tasks in an environment, wherein the user may be able to utilize different formats taken from the conventional platforms. Therefore, Lynn does not teach a data set and process definition that are selected by the user, but a hierarchy that is put into place to shield the user from interacting between the different platforms. In other words, while the levels recited in claim 4 enable the user to monitor workflow in a single application to navigate between activities whilst operating on a data set, the levels in Lynn are used to bring together different formats across an enterprise, and the user does not specify a particular process definition but relies on an intermediary level to make different formats compatible. There is nothing in the teachings of Lynn that provide a linkage between particular activities, e.g. a user selectable process definition, but merely a system to bring together applications across an enterprise in a single environment.

As indicated above, Lynn is concerned with enterprise-wide workflow, whilst claim 4 is directed to workflow within a computer application. Therefore, claim 4 is directed to an entirely different use, and as such its implementation and functionality are inherently different than the framework taught by Lynn.

Accordingly, Lynn does not teach a process level, nor a sub-process level as recited in claim 4. In particular, Lynn does not teach a user selectable process definition for navigating between certain activities associated therewith, at a sub-process level. Moreover, Lynn is concerned with enterprise workflow, whilst claim 4 is directed to workflow in a computer application. As such, Lynn's system operates in an entirely different manner to provide an entirely different result.

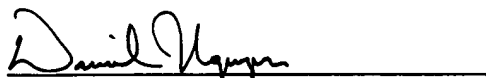
Applicant respectfully submits that amended claim 4 clearly and patentably distinguishes over Lynn, and as such, Lynn cannot anticipate claim 4. Claims 5-32, being ultimately dependent on claim 4, are also believed to distinguish over Lynn.

Claim 33 is amended in a manner similar to claim 4, and operates according to the principles thereof. Therefore, claim 33 is also believed to distinguish over Lynn. Claims 34-51, being ultimately dependent on claim 33, are also believed to distinguish over Lynn.

In view of the foregoing, Applicant respectfully submits that claims 4-51 clearly and patentably distinguish over Lynn, and as such are in condition for allowance.

Applicant requests early reconsideration and allowance of the present application.

Respectfully submitted,



Daniel G. Nguyen
Attorney for Applicant
Registration No. 42,933

Date: November 1, 2005